1976

A microprocessor-controlled data acquisition system for the Federal Scientific Model UA-500-1 ubiquitous spectrum analyzer.

Sauntry, Thomas Frederick

Monterey, California. Naval Postgraduate School

A MICROPROCESSOR-CONTROLLED DATA ACQUISITION SYSTEM FOR THE FEDERAL SCIENTIFIC MODEL UA-500-1 UBIQUITOUS SPECTRUM ANALYZER

Thomas Fredrick Sauntry

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

A Microprocessor-Controlled Data Acquisition System for the Federal Scientific Model UA-500-1 Ubiquitous Spectrum Analyzer

by

Thomas Fredrick Sauntry

September 1976

Thesis Advisor: Thomas M. Houlihan

Approved for public release; distribution unlimited.

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) A Microprocessor-Controlled Data Acquisition System for the Federal Scientific Model UA-500-1 Ubiquitous Spectrum Analyzer		5. TYPE OF REPORT & PERIOD COVERED Master's Thesis; September 1976 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Thomas Fredrick Sauntry		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		12. REPORT DATE September 1976 13. NUMBER OF PAGES 74
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office) Naval Postgraduate School Monterey, California 93940		15. SECURITY CLASS. (of this report) Unclassified 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE
16. DISTRIBUTION STATEMENT (of this Report) Approved for public release; distribution unlimited		
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		
18. SUPPLEMENTARY NOTES		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) microprocessor data acquisition spectrum analyzer microprocessor-control		

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

A microprocessor-controlled data acquisition system is described. The data acquisition system was developed to facilitate the acquisition and analysis of frequency data processed by a Federal Scientific Model UA-500-1 Ubiquitous Spectrum Analyzer. A control program in an Intellec 8 microprocessor is used to control the rate at which spectrum data are output from the spectrum analyzer. An interface circuit converts the data from serial to parallel format and then latches the data for

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-014-6601 |
(Page 1)

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

20. (cont.)

input by the microprocessor. The microprocessor converts the data from binary format to binary-coded-decimal and then outputs the data on cassette magnetic tape. The spectrum data on cassette magnetic tape can subsequently be read back by the microprocessor and then punched on paper tape in a format suitable for input into the IBM-360 computer system.

A Microprocessor-Controlled Data Acquisition System for the Federal Scientific Model UA-500-1 Ubiquitous Spectrum Analyzer

by

Thomas Fredrick Sauntry
Lieutenant, United States Navy
B.S., United States Naval Academy, 1969

Submitted in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE IN AERONAUTICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL

September 1976

ABSTRACT

A microprocessor-controlled data acquisition system is described. The data acquisition system was developed to facilitate the acquisition and analysis of frequency data processed by a Federal Scientific Model UA-500-1 Ubiquitous Spectrum Analyzer. A control program in an Intellec 8 microprocessor is used to control the rate at which spectrum data are output from the spectrum analyzer. An interface circuit converts the data from serial to parallel format and then latches the data for input by the microprocessor. The microprocessor converts the data from binary format to binary-coded-decimal and then outputs the data on cassette magnetic tape. The spectrum data on cassette magnetic tape can subsequently be read back by the microprocessor and then punched on paper tape in a format suitable for input into the IBM-360 computer system.

TABLE OF CONTENTS

LIST OF FIGURES

I. INTRODUCTION

A. PURPOSE OF THE REPORT

The purpose of this report is to present information required to understand the development and operation of a microprocessor-controlled data acquisition system designed to facilitate the acquisition and analysis of frequency data processed by a Federal Scientific Model UA-500-1 Ubiquitous Spectrum Analyzer.

This report describes the reasons for developing the system, the objectives of the system, how these objectives were met, and the operation of the system.

B. ORGANIZATION OF THE REPORT

Section II of this report describes the reasons for developing the data acquisition system and the desired objectives of the system.

Section III gives a detailed description of the major components that make up the system and the operation of these components within the system.

The procedures required for a user to operate the system are covered in Section IV. This section describes system preparation, initialization, and operation.

Section V covers the testing and performance of the completed system.

Section VI contains proposals on future use and expansion of the system and suggested modifications to the computer programs used by the system.

The final portions of the report contain appendices on related system components and programming procedures, listings of the computer programs, and references.

II. OBJECTIVES

A. MOTIVATION FOR DEVELOPING THE DATA ACQUISITION SYSTEM

At the Naval Postgraduate School, the Federal Scientific Model UA-500-1 Spectrum Analyzer is used for several applications that require a real-time, continuous spectrum analyzer. One of the primary applications for this spectrum analyzer is on-board the oceanographic research vessel R/V ACANIA, where it is used for analysis of atmospheric turbulence data.

The Model UA-500-1 Spectrum Analyzer has many desirable features for such applications, but it is somewhat lacking in means of displaying spectrum data after analysis. Output jacks on the front panel, which provide an analog signal representing the spectrum data, can be used to furnish an input to an oscilloscope or a plotter. These forms of data output give a visual display of the spectrum data but can be used for subsequent data reduction and analysis only after manual conversion of the data to a numerical format.

The Model UA-500-1 Spectrum Analyzer also has rear panel multipin connectors that can be used to interface with external equipment. These connectors can be used to control and sense the Analyzer's operating modes, or to receive and transmit spectrum data in analog and digital form.

The digital form of the spectrum data output provides an output that can be used for accurate and permanent records

of the spectrum data.

The purpose of the data acquisition system is to control the presentation and transmission of the digital spectrum data output, and to record the spectrum data for subsequent data analysis.

B. DESIRED FEATURES OF THE DATA ACQUISITION SYSTEM

It was desired that the data acquisition system utilize a microprocessor as the system's primary component. Microprocessors were specifically developed for use as low-cost alternatives to large computers for small-system control and implementation. A microprocessor could be used to control a system that would then be compact and largely automatic in its operation.

It was also desired that the system have the ability to record data in a permanent format that could be used to input the data into the IBM-360 computer system for further analysis.

C. SYSTEM OBJECTIVES

The primary objective of any data acquisition system is to first acquire the data. A suitable computer program and interface circuitry had to be devloped to control and input spectrum data being displayed by the Model UA-500-1.

The system had to be able to record the spectrum data in a permanent format in order that the data be available for subsequent analysis. Cassette magnetic tape was chosen as

the recording medium for this purpose.

The method chosen to present the data for input into the IBM-360 computer system was to punch the data on paper tape in a format suitable for input by the CP/CMS operating system.

Suitable interface circuitry and computer programs had to be developed to meet these objectives.

III. DESCRIPTION OF THE DATA ACQUISITION SYSTEM

A. SIGNAL FLOW IN THE SPECTRUM ANALYZER

The method used to interface the Intellec 8 microprocessor with the Model UA-500-1 Spectrum Analyzer was determined primarily by how the Analyzer processed input data for analysis and display. A brief description of that process is given here so that the method and the performance required of the interface system will be better understood.

1. Vertical Signal Analysis

The input signal to the Analyzer is sampled at a rate determined by the selected frequency analysis range. Data samples from the Analyzer's input are fed into the input analog-to-digital converter, which converts the input signal into digital form. The digitized data samples are then entered into a recirculating digital memory. Samples emerging from the memory output are fed back into the memory input and are sequenced so that all samples are kept in the order in which they arrived.

The recirculation rate of the memory is much higher than the frequency at which the input signal is sampled. Thus, the time period between the emergence of two successive samples at the output of the memory is much shorter than the time period between their being sampled at the input.

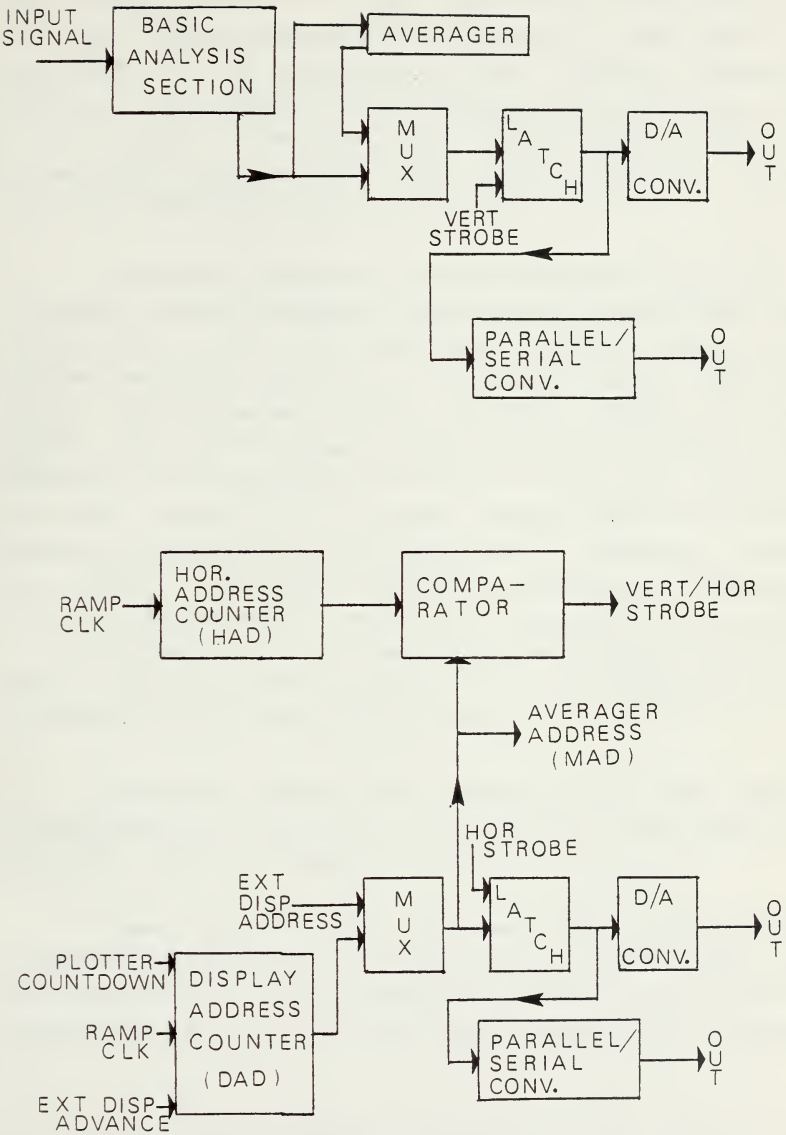

Figure 1 - SIGNAL FLOW IN THE MODEL UA-500-1 SPECTRUM ANALYZER

This signal speed-up obtained from high-speed memory recirculation results in the contraction of the signal's time base. This compression of the signal's time base is equivalent to an expansion of the signal's frequency spectrum. For example, if the time required for a sine wave to go through a complete cycle is reduced by a half, the frequency has been doubled. This signal speed-up allows real-time, continuous signal analysis.

The digital samples in memory are then fed into a high-speed digital-to-analog converter and reconverted into analog form. This time-compressed analog signal is filtered, and then applied to a suppressed carrier modulator signal. The suppressed carrier modulator signal is then applied to a fixed frequency crystal filter. By varying the frequency of the suppressed modulator carrier, the Analyzer can sweep through the analysis range and 500 distinct frequency components of that range will be produced during each sweep. The carrier signal is stepped up in frequency at a 10 kilohertz rate. The spectrum signal is then sampled at a 10 kilohertz rate by a sample and hold circuit, and the complete spectrum of 500 components is thus transmitted from the memory in a 50 millisecond period.

The analog spectrum data coming out of the sample and hold stages is converted to an eleven-bit binary word by the output analog-to-digital converter.

These spectrum data represent the instantaneous spectrum being generated by the Analyzer. These data are then fed into the optional Averager circuits. The Averager circuits average these data with prior spectrum data (Figure 1).

Both the averaged spectrum data and the instantaneous spectrum data are applied as inputs to a

two-input multiplexer. The Analyzer's front panel display controls are set by the user to determine which spectrum signal is to be displayed. The multiplexer selects either the averaged or the instantaneous spectrum data and transmits them to the display and output circuits.

2. Timing In The Display And Output Circuits

In the display and output portions of the Analyzer, the basic timing cycles are determined by the circulation rate of the memory circuits. The memory recirculates every 100 microseconds, and a timing pulse labeled the LOAD ENABLE pulse is generated at the beginning of each circulation period. The LOAD ENABLE pulse controls the timing in the display and cutput circuits.

3. Horizontal-Display Circuits

The horizontal display circuits control how the vertical data signals are presented to the output circuits for display.

Two different counters are used in the horizontal-display circuits. These counters are used to assign an address to each spectrum component and to determine when that component will be displayed.

The first counter is the horizontal address counter, or HAD counter. The HAD counter uses the LOAD ENABLE pulse for its input. Every LOAD ENABLE pulse indicates that a new frequency spectrum point is being transmitted from the memory. The HAD counter continuously counts from one to five hundred, and its output thus represents the horizontal address for each spectrum component in the vertical data

signal.

The second counter in the horizontal-display circuits is the display address counter, or DAD counter. The DAD counter determines the sweep rate at which the spectrum data are presented to the output circuits.

There are several inputs to the DAD counter which can control the rate at which it is stepped. In the normal FAST OSCILLOSCOPE mode of operation, the DAD is stepped at the same 10 kilohertz rate as the HAD counter. In this mode the complete spectrum signal is presented to the output circuits in a 50 millisecond period. The output is then blanked for a 50 millisecond period to allow the oscilloscope time for retrace.

In the PLOTTER mode of operation, the DAD counter is stepped at a 20 hertz rate. This rate is slow enough to accommodate the relatively slow speed of an external plotter.

There are also two ways that the DAD counter can be controlled by external signals.

In the first method, an external DISPLAY ADVANCE pulse is applied through Pin No. 74 on the rear panel INPUT/OUTPUT connector J1. Each pulse will advance the DAD counter one step.

The second method of externally controlling the display address is to input the desired address in binary form as digital pulses applied to the appropriate eleven pins on connector J1. These pins are listed in Appendix E.

The display address is then routed to the Averager to supply a proper address for spectrum averaging

operations. This address is then the memory address, or MAD.

The MAD and the HAD are then fed into a comparator. The HAD is continuously cycling through the five-hundred data addresses at a rate of 10 kilohertz. Each address corresponds to the data component then being transmitted from the sample and hold stages. The MAD will be stepping through the desired display addresses at the same rate that the DAD counter is being stepped.

Whenever the MAD and the HAD are equal, a vertical and horizontal data strobe signal is generated by the comparator. This strobe signal is used to sequence the data into the output circuits.

4. Output Circuits

The spectrum data output from the first multiplexer is applied to a latch, which is strobed at the appropriate time by the vertical data strobe signal generated by the horizontal-display circuits. Each time that the latch is strobed, it is updated with a new vertical spectrum data component.

The data in the latch are available as output signals in both digital and analog form. Part of the output circuitry converts the signal back into analog form. It can then be used as the input signal for an external oscilloscope or plotter.

The digital data are also converted to a serial format and shifted out on Pin No. 51 on the rear panel INPUT/OUTPUT connector J1. Each spectrum component is presented as a binary word of twelve bits. The spectrum

data from the Averager are always normalized to the full twelve bits. However, the instantaneous spectrum from the Analyzer is normalized to eleven bits. The twelfth and last bit will always be a logical zero.

The latched data in the output circuits are thus transmitted once every 100 microseconds. The rate at which the spectrum data points being transmitted at the output are stepped through the entire five hundred spectrum components of the analysis range is determined by the rate at which the DAD counter is being stepped, either internally or externally.

5. Rear Panel INPUT/OUTPUT Connector J1

The rear panel INPUT/OUTPUT multipin connector J1 on the Model UA-500 Spectrum Analyzer enables the Analyzer to be interfaced with external equipment. Input and output signals on this connector are TTL compatible. A logical zero (0) input or output voltage must be in the range from 0 to +0.5V. A logical one (1) must be within the range from +2.5 to +5.0V. Input and output pulse widths are nominally 3.2 microseconds.

This section covers some of the more important input and output signals present on connector J1 that are used in the interface system.

The VD OUT (Vertical Data Output) pulse output on Pin No. 51 is the digital spectrum output. It is a twelve bit, serial, binary word which occurs once every 100 microseconds (Figure 2). One word is present for each of the five hundred spectrum components, and the horizontal-display circuitry controls which component is being displayed. Twelve bits are transmitted for Averager

data. Eleven bits are transmitted for Analyzer data, the twelfth bit from the Analyzer always being a logical zero.

The HD OUT (Horizontal Data Output) pulse output on Pin No. 51 is the digital address word which represents the spectrum horizontal address associated with the data word being presented at Pin No. 50. This word also appears once every 100 microseconds. It consists of twelve bits in a serial format. Nine bits are used for the data address, with bits 9, 10, and 11 always being a logical zero.

The BIT CLOCK pulse output on Pin No. 57 is necessary for the serial-to-parallel conversion of spectrum data in the interface circuits. This digital word always has twelve logical-one bits. The pulse repetition rate is 312 kilohertz. Each bit is transmitted synchronously with the HD OUT and VD OUT outputs so that the presence of a logical zero (absence of a pulse) can be detected in these outputs. The first pulse represents the most significant bit position of data transmission.

The WORD CLOCK pulse output on Pin No. 53 is also used for serial to parallel conversion of spectrum data. This pulse occurs once for each data word and is at a logical one level when data bits are being transmitted. It is at a logical zero level when data are not being transmitted.

The EXT ADD ENABLE level input on Pin No. 1 is used to select internal or external control of the display address mode. External display address control is enabled when this input is at a logical zero level.

EAD 0 through EAD 10 are external address inputs on Pin Nos. 4, 8, 12, 15, 18, 22, 25, 28, 31, 34, and 37. These inputs permit external control of the display address

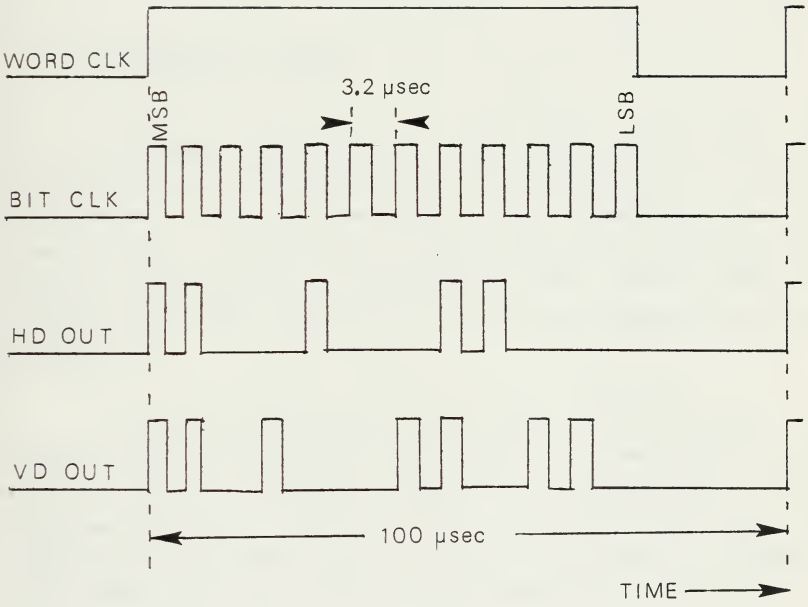

Figure 2 - TIMING DIAGRAM FOR DIGITAL SPECTRUM OUTPUT

when this mode is enabled by the EXT ADD ENABLE input. The display address consists of nine binary bits, EAD 0 through EAD 8. Bit EAD 9 can be used to select Averager memory A (logical zero), or memory B (logical one). Bit EAD 10 is used for optional add-on memories.

B. INTERFACE TIMING AND CONTROL

1. Intellec 8 Timing

All CPU operations on the 8008-1 CPU chip are determined by a two-phase clock. This clock has a basic time state of 2.5 microseconds. The execution time of instructions in the 8008-1 takes from three to eleven time states, depending on the instruction class. This gives an instruction cycle time of 7.5 to 27.5 microseconds for the Intellec 8.

When the instruction cycle time of the Intellec 8 is compared with the spectrum data digital transmission rate of 312 kilohertz in the Model UA-500-1 Spectrum Analyzer, it is apparent that the Intellec 8 is at least an order of magnitude too slow for it to be capable of directly interfacing with the Analyzer for receiving spectrum data.

For this reason, an interface system was developed to act as a buffer between the Intellec 8 and the Model UA-500-1 Spectrum Analyzer.

2. Interface System Requirements

The primary requirements of the interface system are

that it must be able to control both the rate and the order in which spectrum data are transmitted from the Analyzer; it should be able to convert the digital spectrum data from serial to parallel format for greater efficiency from the Intellec 8 in inputting these data; and it must latch the data until the Intellec 8 has read them in.

The requirement to control the presentation of spectrum data is absolutely necessary if the data are to have any meaning for subsequent use.

Converting the data from serial to parallel format is not strictly necessary, but the speed and efficiency of the Intellec 8 is much greater when data are presented in parallel format at the appropriate input ports.

The requirement that the data be latched until the Intellec 8 has read it in is dictated by the asynchronous nature of the interface system.

3. Spectrum Address Control By The Interface

Externally controlling the display address in the Analyzer by means of direct input of the desired display address was chosen as the method to be used by the interface system. This method has the virtue of allowing specific control of the display address at all times. The address displayed by the Analyzer is always the address that the Intellec 8 is supplying through EAD 0 through EAD 8.

C. INTERFACE CIRCUIT BOARD

1. Circuit Description

The interface circuit board performs the conversion of spectrum data from serial to parallel format and latches these data for input to the Intellec 8.

The circuit is constructed on a 4 inch by 4.5 inch perforated circuit board with a 22-pin card-edge connector for circuit input and output signals (Appendix E).

Inputs to the circuit are the VD OUT, BIT CLOCK, and WORD CLOCK signals from the Analyzer; and the Clock Flip-Flop Set and Reset signals from the Intellec 8. The circuit outputs are data bits 0 through 11 to the Intellec 8.

2. Serial To Parallel Data Conversion

IC2, IC3, and IC4 (Figure 5) are N74195 4-bit parallel access, shift registers. VD OUT data present at the input pins 2 and 3 of IC2 are shifted in serially by a clock pulse on pin 10. As the data are shifted in, they are latched into the parallel output pins 15, 14, 13, and 12. Output D on pin 12 becomes the input to IC3. In the same way, IC4 is connected in series with IC3. Thus, after twelve clock pulses, the data word is converted to parallel format and latched into the twelve output pins.

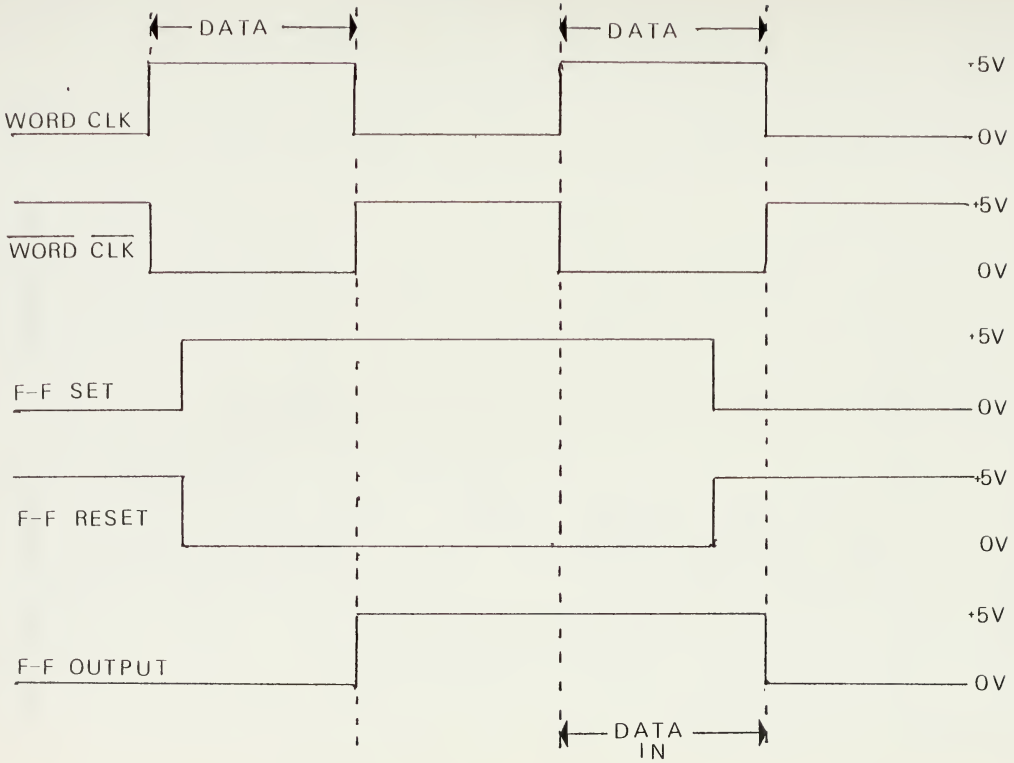

Figure 3 - TIMING DIAGRAM FOR THE INTERFACE CIRCUIT BOARD

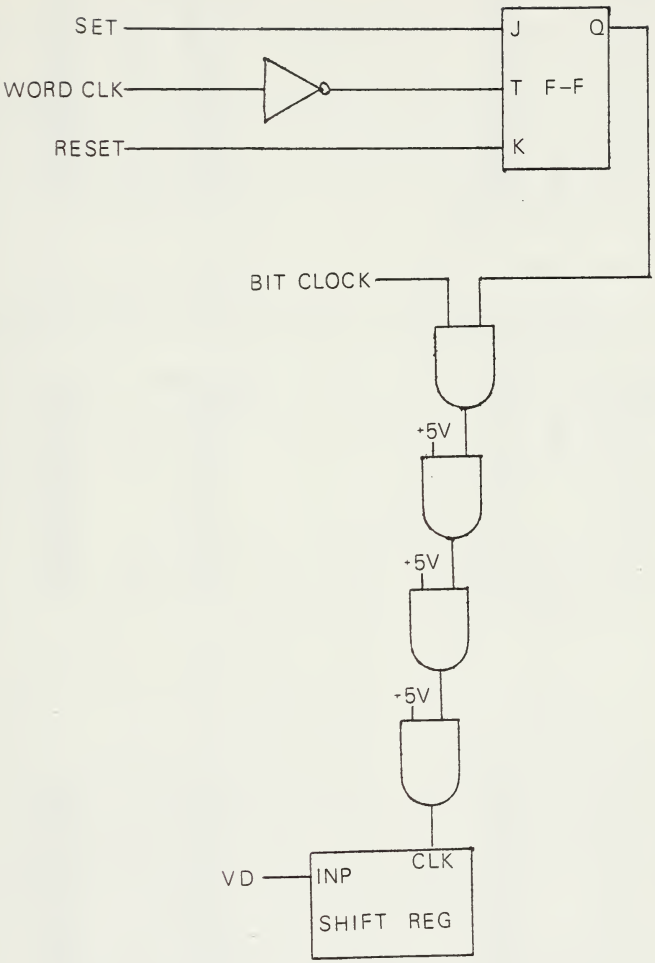

Figure 4 - LOGIC DIAGRAM FOR THE INTERFACE CIRCUIT BOARD

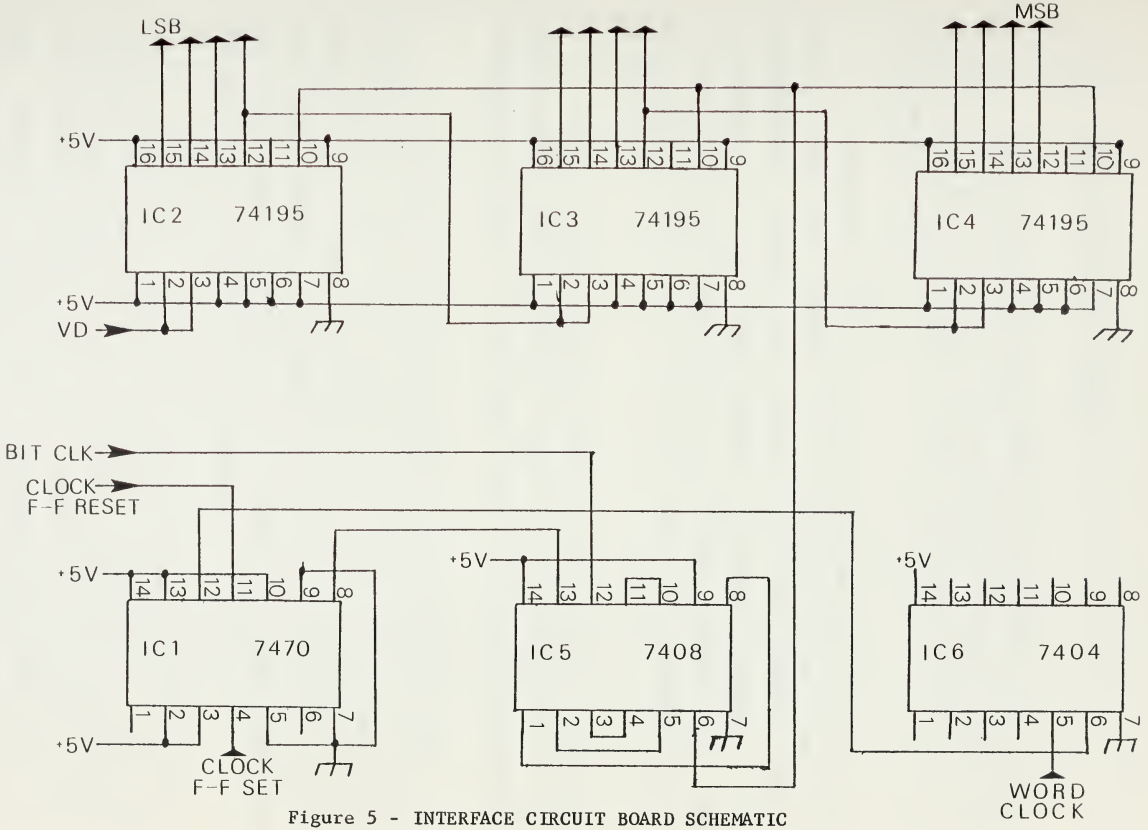

Figure 5 - INTERFACE CIRCUIT BOARD SCHEMATIC

3. Clock Delay And Gating

IC1, IC5 and IC6 are used for the delay and gating of the clock signal. IC1 is an N7470 J-K, gated flip-flop; IC5 is an N7408 quadruple, 2-input AND gate; and IC6 is an N7404 hex inverter.

The N74195 shift register requires that data at the serial input be present for a minimum set-up time of 15 nanoseconds before the arrival of the clock pulse. Therefore, the BIT CLOCK signal is passed through four AND gates prior to its application to the shift registers. The four gates delay the BIT CLOCK signal by approximately 70 nanoseconds.

IC1 and IC6 provide the logic necessary to properly gate the BIT CLOCK signal in order to prevent its being constantly applied to the clock pins of the shift registers.

The BIT CLOCK and the VD OUT data signals from the Analyzer are transmitted every 100 microseconds. If the BIT CLOCK signal were not gated, it would be repetitively shifting the VD OUT data through the shift registers during each data transmission from the Analyzer. If the Intellec 8 tried to input the data at a time when the data were being shifted, it would be subject to serious errors.

Therefore, the BIT CLOCK must be disabled when the Intellec 8 is performing a data input operation. The BIT CLOCK signal must also be disabled only when a valid data word has been shifted into the registers. If the BIT CLOCK were disabled during the transmission of a data word, the word would be incomplete. Therefore, the BIT CLOCK signal must only be disabled between data transmission periods.

The timing diagram in Figure 3 and the logic diagram in Figure 4 illustrate how proper gating of the BIT CLOCK signal is achieved.

The WORD CLOCK signal from the Analyzer is inverted by IC6. Thus, when data transmission starts, the inverted WORD CLOCK signal goes to a logical zero level. When data transmission is ended, the inverted WORD CLOCK signal goes to a logical one level.

When data are to be shifted into the shift registers, the program sets the Clock Flip-Flop Set signal from the Intellec 8 to a logical one level. The Clock Flip-Flop Reset signal is set to a logical zero level. When these signals are applied to the J-K flip-flop, the next logical one transition of the inverted WORD CLOCK signal sets the flip-flop output to a logical one level. When the J-K flip-flop is set, its output enables the BIT CLOCK signal at the first AND gate and data is shifted into the shift registers.

To disable the BIT CLOCK signal, the control program in the Intellec 8 sets the Clock Flip-Flop Reset signal to a logical one level, and the Clock Flip-Flop Set signal to a logical zero level. When the next data transmission is ended, the inverted WORD CLOCK goes to a logical one level and resets the J-K flip-flop. This causes the output of the flip-flop to go to a logical zero level and disable the BIT CLOCK signal at the first AND gate.

D. COMPUTER PROGRAMS

The two computer programs listed on Pages 65 to 72 are used to control and implement the data acquisition system.

The four PL/M procedures listed on Pages 65 to 68 provide control of the data interface between the Intellec 8 and the spectrum analyzer, convert the binary data into binary-coded-decimal (BCD) format, and then write the data on cassette magnetic tape through the Memodyne Digital Cassette Recorder.

The machine-language program on Pages 69 to 72 is used to output a block of data in Intellec 8 memory on punched paper tape in a format suitable for input into the IBM-360 computer system.

Both programs have numerous comment fields which detail the operations being performed.

1. PL/M Program

This program consists of four procedures, or subroutines. Each procedure has been written so that it is independent of the rest of the program. All parameters used to pass information between procedures are formal parameters, and all variables are declared inside of each procedure. Thus, changing a parameter inside of a procedure does not affect that parameter in the calling procedure or program. Variables in the main program and in other procedures can have the same PL/M identifier as a variable inside of a procedure and the two variables will be completely independent. These procedures have been written in this manner so that they may be used in any program without the need for major revision.

a. Procedure DATA$FETCH

Procedure DATA$FETCH controls the data interface

between the Intellec 8 and the spectrum analyzer.

The nine external address inputs to the Analyzer are provided by the eight bits of OUTPUT PORT (07) and the third bit of OUTPUT PORT (06). The first bit of OUTPUT PORT (06) is used to enable the EXTERNAL ADDRESS mode of the Analyzer. The fourth and fifth bits of OUTPUT PORT (06) are used to route the Clock Flip-Flop Set and Reset signals to the interface circuit board.

INPUT PORT (06) and INPUT PORT (07) are used to input the spectrum data from the interface circuit board.

In the procedure, the program external address counter, BIN$NUMBER, is first initialized to zero. This counter will be stepped from 0 to 499 decimal, and determines the horizontal address of the spectrum data component to be transmitted.

The program external address is applied as a signal to the Analyzer and a five hundred microsecond delay is invoked to allow the Analyzer sufficient time to display the desired data component. The Clock Flip-Flop Set line is then activated to enable the interface circuit board to shift the data into the shift registers. Another five hundred microsecond delay is invoked to allow time for the data to shift into the registers. The Clock Flip-Flop Reset line is then activated to disable the shift circuitry. Another delay ensures time for the shift circuitry to be disabled and the data are then read by the Intellec 8. The external address counter is then stepped and the process is repeated until five hundred spectrum data components have been read by the Intellec 8.

The data are stored as an array of 1,000 consecutive bytes in memory. Procedure DATA$FETCH then

calls Procedure MAG$TAPE$OUT. The parameter in this CALL statement is a pointer to the address of the first element of the data array.

b. Procedure MAG$TAPE$OUT

Procedure MAG$TAPE$OUT converts each two-byte word of data to BCD and then writes the BCD data word on cassette magnetic tape. Procedure MAG$TAPE$OUT calls upon Procedure DELAY for controlled time delays and calls upon Procedure UN$HEXER to convert data from binary to BCD.

The procedure first advances approximately twelve seconds of blank tape in the recorder. This ensures that a new cassette will be advanced off of the transparent leader. This blank portion of the tape also serves to delineate multiple files on the tape.

Each spectrum data component is converted into two eight-bit bytes. Each byte is composed of two four-bit BCD numbers representing the decimal value of the spectrum component. Spectrum data are represented as twelve-bit binary numbers and therefore have a maximum value of 4,096 decimal. The data are written on the cassette as a 1,000 byte file with no gaps between data. No file identifiers or special symbols are produced by this program.

2. Paper Tape Punch Program

This machine-language program takes a block of data from Intellec 8 memory and outputs the data on punched paper tape in a format that can be read directly by the CP/CMS system on the IBM-360 computer.

Two arguments are expected by the program. The beginning address of the block of memory to be punched must be loaded into memory addresses 2FF0 and 2FF1 hexadecimal. The high-order address byte must be in memory address 2FF0 hexadecimal. The final memory address, plus two, of the block to be punched must be loaded in the same format into memory addresses 2FF2 and 2FF3 hexadecimal.

Two subroutines that are part of the Intellec 8 Resident System Monitor program are used to punch the paper tape. The System Monitor subroutine at memory address 3EB5 hexadecimal expects a character in Register B as an argument. This character is punched on paper tape exactly as it is found.

The System Monitor subroutine at address 3E7B hexadecimal expects a character in Register A as an argument. This character is converted to ASCII Code and is punched as two bytes on paper tape.

The data are punched on paper tape in the following format:

(1) The tape is punched in 8-level ASCII Code.

(2) There are no "null" characters. Spaces are not null characters.

(3) Each data record consists of twenty consecutive bytes from memory. These twenty bytes represent ten spectrum data components. These data will be punched as ten groups of four characters each. Two spaces will precede each group of four characters. The last two characters in each record will be an "X-OFF" punch followed by a space.

(4) The final record will be terminated by three "X-OFF" punches. The arguments in memory addresses 2FF0 through 2FF3 are used by the program to determine the beginning and the end of the data block in memory.

IV. USER OPERATION OF THE DATA ACQUISITION SYSTEM

A. EQUIPMENT PREPARATION

The Model UA-500-1 Spectrum Analyzer, the Intellec 8 computer, the Memodyne Digital Cassette Recorder, and the interface circuit board must be connected to each other as described in Appendix E. Connections should be checked for proper installation and integrity. Each piece of equipment should have power applied and be operating correctly.

The Analyzer should have its controls set for operation as described in Reference 1. Data should be set for FAST OSCILLOSCOPE mode display. Either instantaneous or averaged spectrum data can be chosen to be displayed. The proper memory of the Averager should be selected if averaged data are to be displayed. In the EXTERNAL ADDRESS mode of operation, the digital output on the rear panel connector is not affected by most of the front panel controls.

The Intellec 8 should be under the control of the System Monitor program until data are to be transferred. The PL/M control program object code should be loaded in memory.

The Memodyne Digital Cassette Recorder should have a cassette in place, with the READ/WRITE head mechanism locked in the operating position. The cassette to be used should be rewound to the leader and must be a certified digital quality tape.

B. PROGRAM EXECUTION

When all equipment is operating correctly and is properly connected, the control program is executed by a "GXXXX" command to the System Monitor, where XXXX represents the hexadecimal address of the first program instruction.

The transfer of the five hundred spectrum data components to the Intellec 8 should take from five to ten seconds. At this time the cassette in the Memodyne Recorder should begin advancing. The cassette will advance for approximately twelve seconds before data are written onto the magnetic tape. At this time, a faint chattering sound will be heard from the stepping motor in the tape transport system. This noise is to be expected and indicates proper operation.

Transfer of the data to the cassette magnetic tape should take from fifteen to forty-five seconds. The speed at which data are written on the cassette magnetic tape is largely determined by the relative size of the data components which have to be converted from binary to BCD by the program.

When all data have been transferred to the cassette magnetic tape, the recorder will stop advancing and control of the Intellec 8 will be returned to the System Monitor program. At this time the cassette may be removed from the Memodyne Recorder, or left in place if more signals are to be analyzed by the Model UA-500-1 Spectrum Analyzer. Subsequent data runs can be made by repeating the above procedures. However, the tape should not be rewound if more than one data file is to be recorded. Multiple files on the tape will be separated by a twelve-second gap of blank tape.

C. RECOVERING DATA FROM THE CASSETTE MAGNETIC TAPE

To recover data that have been recorded on cassette magnetic tape, these data must be read into the Intellec 8 by the Memodyne Recorder.

A short machine-language program for reading cassettte magnetic tapes on the Memodyne Recorder has been previously written by LCDR J. R. Plunkett. This program has been recorded on a PROM and is permanently installed in memory on the Intellec 8.

To read in recorded data by using this program, the Memodyne Recorder and the Intellec 8 must be connected as before with the 25-pin cannon connector. A block of memory must be chosen for storage of the data as it is read into the Intellec 8. For example, a data file of 500 spectrum components will occupy 1,000 consecutive bytes of memory. This block of memory must be chosen such that the highest memory address is less than 2F00 hexadecimal.

The beginning memory address of this block of memory is then loaded into memory addresses 2FF0 and 2FF1 hexadecimal. The high-order byte of the memory address is to be placed in memory address 2FF0 hexadecimal. The number of bytes of data to be transferred is then placed into memory addresses 2FF2 and 2FF3 hexadecimal. This number is to be in hexadecimal, with the high-order byte in memory address 2FF2 hexadecimal.

For example, if 1,000 (3E8 hexadecimal) bytes of data are to be read into a block of memory starting at address 0400 hexadecimal, the contents of memory addresses 2FF0 through 2FF3 would be the following: 04-00-03-E8. These data would then be stored in memory addresses 0400 through

07E8 hexadecimal.

This program is executed by a "G3030" command to the System Monitor. After execution, the cassette should be seen advancing, and a slight chattering sound will again be heard as the data are read by the Memodyne Recorder. When the appropriate amount of data has been read, the cassette motion will stop and control of the Intellec 8 will return to the System Monitor program.

D. PUNCHING DATA ON PAPER TAPE

After data have been recovered from the cassette magnetic tape and are in memory storage in the Intellec 8, they can then be punched on paper tape.

The Intellec 8 system configuration for the Intellec 8 computer in Rm. 017 of Halligan Hall includes a Model AN/UGC-59A teletype and paper-tape punch console. This unit is connected with the Intellec 8 computer and punching operations are controlled by the System Monitor program. The machine-language paper-tape punch program listed on Pages 69 to 72 utilizes the appropriate control subroutines of the System Monitor program . The teletype/paper-tape punch unit is operated by turning the power switch to the ON LINE position. The teletype/punch mode switch can be used in any mode that enables the paper-tape punch, but the operator must manually enter C/R and LF commands at the keyboard if the teletype is printing as the data are punched. By operating in the TTR mode the teletype will be disabled during punching operations.

The machine-language paper-tape punch program has three "No Operation" commands in addresses 2F66, 2F67, and 2F68

hexadecimal. A Carriage Return/Line Feed command could be inserted here as "46-95-3E" hexadecimal commands. With this program modification, the teletype would print the data as they are being punched and a record could be made of the data. However, if this change is made to the program, the Carriage Return/Line Feed commands would have to be deleted by use of the CP/CMS system Editor when the data are read into the IBM-360.

The paper-tape punch program is then loaded into the Intellec 8. The starting memory address of the data to be punched is loaded into memory addresses 2FF0 and 2FF1 hexadecimal. The last data address plus two is loaded into memory addresses 2FF2 and 2FF3.

For example, if the 1,000 bytes of data read in by the previous example were to be punched , the contents of memory addresses 2FF0 through 2FF3 would be the following: 04-00-07-EA.

When this information has been properly entered into the Intellec 8 and the teletype has been readied, a blank leader can be punched on the paper tape by an "N" command to the System Monitor program. After this leader has been punched, the data are punched by executing the program with a "G2F00" command to the System Monitor. Another leader can be punched on the tail-end of the tape after the data have been punched.

E. INPUT OF PAPER TAPE DATA TO THE IBM-360

Data on paper tape can be read into the IBM-360 by following the procedures in Reference 2. The data punched by the paper tape punch program have the correct format for

entry into the CP/CMS system data files.

To read data out of the files after these data have been read by the CP/CMS system requires an appropriate FORTRAN FORMAT statement. A suggested FORMAT type would be FORMAT (10I6). The data are arranged as ten four-digit values per record, with two spaces preceding each group.

V. SYSTEM PERFORMANCE AND TESTING

A. PERFORMANCE OF THE DATA INTERFACE SYSTEM

The data interface system between the Intellec 8 and the Model UA-500-1 Spectrum Analyzer was tested by using the HD OUT (horizontal data output) signal on Pin No. 50 to represent the VD OUT (vertical data output) on Pin No. 51. By using the digital display address output in place of the spectrum data output as an input to the interface circuit board, an accurate appraisal of the system could be made. If the system were operating correctly, the data recorded on the cassette magnetic tape by the Memodyne Recorder would be a decimal format of the spectrum component addresses. These addresses are supplied by the control program and should progress from 0 to 499 decimal.

When the data interface system was tested in this manner, the output from the magnetic tape was not as expected. A large proportion of the output numbers were not in the correct range and were not ordered correctly. However, the output numbers did have a recurrent type of error pattern in that two consecutive numbers would often be equal.

Test runs using known data in memory showed that the portions of the control program which converted the data to BCD and then wrote the data on cassette magnetic tape were operating correctly.

A short machine language program was written to further test the data interface circuit board and the operation of the spectrum analyzer. This program would output one selected display address and the appropriate control signals. The single data component would be read into the Intellec 8 and the control signals would remain applied.

An oscilloscope applied to the appropriate data lines showed that the Model UA-5000-1 Spectrum Analyzer was the source of the errors.

The Analyzer did not appear to be operating correctly in two aspects of its data display and output. The least significant bit of the display address counter was not corresponding to the external address input signals. This bit remained at a logical zero level for all inputs. Therefore, only even number display addresses were being transmitted.

The second malfunction in the Analyzer was more serious in its effect on the data transmission. A dual-trace oscilloscope applied to the BIT CLOCK and HD OUT output signals showed that the two signals were not being transmitted synchronously. The BIT CLOCK signal was lagging the HD OUT signal by almost 3 microseconds. The BIT CLOCK signal was being transmitted in such a way that it was often-times being transmitted between the corresponding pulse of the display address output and the pulse immediately following.

This resulted in spurious clock signals being applied to the shift register circuitry of the interface circuit board. If the clock pulse arrived coincidentally with the corresponding data pulse, the data pulse was being correctly clocked into the circuitry. If the clock pulse arrived between data pulses, neither was clocked into the circuitry.

Finally, if the clock pulse arrived coincidentally with the leading edge of the following data pulse it would clock that pulse in early.

These two discrepancies in the Analyzer appeared to be the only causes of error in the data interface.

B. PERFORMANCE OF OTHER SYSTEM COMPONENTS

The control program and interface circuit board appeared to be operating correctly. The Intellec 8 was reading in exactly what was being latched into the interface circuit board outputs, and these data were being correctly converted to BCD and transferred to the cassette magnetic tape.

Paper tapes punched with these data were entered into the IBM-360 by the use of the CP/CMS system. These data were read in correctly and were in the specified format.

VI. CONCLUSIONS AND RECOMMENDATIONS

A. PROPOSED INTERFACE CIRCUIT BOARD MODIFICATIONS

The Model UA-500-1 Spectrum Analyzer used for development and testing of the data acquisition system was returned to the manufacturer for repair and calibration.

A modification to the interface circuit board should be made so that possible discrepancies in the timing relationship between the BIT CLOCK signal and the VD OUT and HD OUT signals will be accounted for by the circuit.

Two possible ways to modify the interface circuit board to accommodate the lag in the BIT CLOCK signal would be to delay the spectrum data signal pulses, or to extend the length of these pulses.

To delay the data pulses, a delay line or an RC delay circuit could be placed in the circuit at a point prior to where the data pulses enter the shift registers. Care would have to be taken to ensure that either the inductance of the delay line or the capacitance of the RC circuit did not excessively round off the leading and trailing edges of the digital data pulse. The delay time from this circuit would have to be carefully adjusted so that the corresponding clock pulse would be clocking the correct data pulse into the shift registers.

To extend the length of the data pulse, a monostable

multivibrator such as the N74121 type could be inserted into the circuit. This type of multivibrator is triggered by an input signal and produces a pulse whose length can be determined by external timing components. The input data pulse could be used to trigger the monostable multivibrator. The length of the output pulse would have to be chosen carefully so that it would be clocked by the lagging clock pulse corresponding to it but not be long enough to be clocked into the circuit by more than one clock pulse.

B. FUTURE SYSTEM EXPANSION

1. Dedicated Stand-Alone Microprocessor

The Intellec 8 was used for the development and testing of this system and is the only microprocessor used. The use of the Model UA-500-1 Spectrum Analyzer on the oceanographic research ship R/V ACANIA makes desirable a separate, small, stand-alone microprocessor in the data acquisition system. Such a microprocessor could be designed as a dedicated controller, used to control only the data interface and the transmission of the data to cassette magnetic tape. The Intellec 8 would then be used only to read the data off the cassette magnetic tape and punch the data on paper tape.

The dedicated stand-alone microprocessor could use the PL/M control program developed for this project as a basis for a larger, more comprehensive program for controlling the data interface. In order to use this program, the microprocessor would have to be one of the many types on the market that make use of the INTEL 8008-1 CPU chip. The use of a different CPU chip would require

different timing considerations and control sequences in the writing of the control program.

The rear panel connectors on the spectrum analyzer carry many input/output signals and control/sense signals that were not used for this project. A larger program, permanently installed in PROM memory in the microprocessor, could easily sense the completion of spectrum averaging modes, annotate the cassette magnetic tape with the frequency range and number of spectra, and store the data as multiple files on the cassette. A dedicated microprocessor-controlled data interface system such as this would be quite compact and largely automatic in its operation.

2. Program Modifications

The PL/M control program was written with the possibility of future expansion in mind. For this reason, the procedures in the program were written to be independent of each other. The major modifications that would have to be made to use this program in a microprocessor other than the Intellec 8 would be in the area of input/output procedures.

The Intellec 8 was designed with complementary inputs and outputs. This means that all inputs and outputs are complemented by the input and output circuitry. If the hexadecimal word "FF" is applied to INPUT PORT (06), the accumulator will receive a "00" hexadecimal when it executes an instruction to input INPUT PORT (06). Likewise, the value "62" hexadecimal in the accumulator would be sensed as "9D" at the output port during an output operation. The System Monitor program takes this feature into account in its input and output subroutines. Data punched on paper

tape are represented exactly as found in memory.

However, the machine-language programs used to read and write data from the Memodyne Digital Cassette Recorder do not automatically complement the data again as they are read. Data read in by the Memodyne Recorder are thus stored in memory as the complement of what are represented on the cassette magnetic tape. The question of whether or not data are in the correct form or have been complemented can be a very confusing one, and care must be taken whenever the Intellec 8 is to be used for input and output operations.

Procedure DATA$FETCH in the PL/M control program was written in such a way that all input and output values are again complemented by the program software. This was done through the use of defining macros to represent input and output instructions. For example, the PL/M instruction to input the data at INPUT PORT (06) is simply INPUT (06). In the string of DECLARATION statements at the beginning of the Procedure, the macro string IN06 was declared to be literally ' NOT INPUT (06)'. Therefore, whenever a value is to be read in at INPUT PORT (06), the macro IN06 is used. In this way, the data read in by the program are always in the same form as applied to the input ports.

In a similar manner, data or control signals that are to be sent to an output port are also complemented by the program. The macro string OUT03 was declared to be literally 'OUTPUT (03)' and the macro string EQ was declared to be literally ' = NOT'. The statement OUT03 EQ 02H will cause the value 02H to be received on the output lines.

Procedure MAG$TAPE$OUT does not use similar macro declarations to complement input and output values. Control signals to the Memodyne Recorder are defined in their complementary form in each output instruction. Data from

memory are also left as is for output transmission to the Memodyne Recorder. The data that are written on cassette magnetic tape are thus the complement of the data as stored in memory.

This method was chosen because the program to read data into the Intellec 8 from the Memodyne Recorder does not complement these data as read. When the data are eventually read back into the Intellec 8, they will be stored in memory in proper form. These data will also be punched on paper tape in proper form, since the System Monitor automatically takes care of this.

The comment fields in the control program indicate which program statements would have to be changed if the program were to be run on a microprocessor other than the Intellec 8. Those comments with a (1) indicate macro declarations that would have to be changed if a different input or output port were to be used. For example, if OUTPUT PORT (01) is to be used, vice OUTPUT PORT (05), only the macro would have to be changed in the program.

Those statements with a comment (2) indicate that the operand of the output statement should be complemented if the microprocessor being used does not have complementary inputs and outputs.

Those statements with a comment (3) indicate that the NOT should be deleted from the macro declaration if the microprocessor being used does not have complementary inputs and outputs.

The final program modification that could be necessary involves the spectrum data input to the interface circuit board. Twelve bits of data are shifted into the shift registers by the BIT CLOCK signal. The most

significant bit of data is the first bit to be shifted in. If the instantaneous spectrum data from the Analyzer is to be transmitted , the last bit will always be a zero. When only eleven bits of data are transmitted, the data value that is read into the Intellec 8 will be twice the magnitude of the true value. This is due to the extra bit in the least significant position. Therefore, data from the Analyzer portion should be shifted one bit to the right by the program before it is put on cassette magnetic tape. Averaged spectrum data from the Averager are accurate. The Averager is generally used for most applications and the program was written for use with averaged spectrum data.

APPENDIX A

UBIQUITOUS SPECTRUM ANALYZER

The Federal Scientific Model U-500 Ubiquitous Spectrum Analyzer is manufactured by the Federal Scientific Corporation. The Model U-500 is a real-time spectrum analyzer which performs signal analysis with 500 spectral-line resolution. It has seventeen selectable analysis ranges, giving a total frequency analysis range of zero to one hundred kilohertz. With the UA-500-1 Analyzer Averager option it can perform spectrum averaging in either sum, exponential, or peak averaging modes.

The Model U-500 is of solid-state construction and makes extensive use of digital circuitry, digital-to-analog/analog-to-digital conversions, and digital signal processing techniques. Continuous, real-time analysis is accomplished by the use of signal storage and speed-up techniques.

Three rear panel multipin connectors allow the Analyzer to be connected to external equipment. These three panels are the INPUT/OUTPUT connector J1, the REMOTE/SENSE CONTROLS connector J2, and the OPTIONS CONTROL connector J3.

Through the use of these rear panel connectors, external equipment can sense or control the Analyzer's operating modes, control the display modes of the spectrum data, or access analog or digital spectrum output.

APPENDIX B

MEMODYNE DIGITAL CASSETTE RECORDER

The Model 173/133 Digital Cassette Recorder, manufactured by the Memodyne Corporation, is a parallel read/write recorder that is used for recording and playback of digital data. The Model 173/133 is easily adapted for use with either the Intellec 8 or the Hewlett-Packard 9830 computer.

The Memodyne Series 100 systems use standard Phillips certified magnetic tape cassettes as their recording media. The Series 100 system consists of modular components and subsystems that can be combined as ten different models. Thus, each model can have the tape transport system and electronics combination most suited for the chosen application.

The Model 173/133 system is made up of the following sub-systems:

1.) Transport Card
2.) Rewind EOT/BOT System
3.) Chassis Card - - 900
4.) Motor Driver - - 903
5.) Write Step - - 902
6.) Read Card - - 901
7.) ASCII Card - - 934

The Model 173/133 is usually used to record and read seven-bit ASCII characters. The unit used for this project

was adapted for use with eight-bit binary characters by the use of an internal jumper cable on Card 934.

In this system, tape motion is controlled by a stepping motor which incrementally advances the tape at a rate of 120 motor steps per inch. This allows the tape motion to be precisely controlled, and one character can be written on each incremental portion of tape. The character incremental recording feature of this system allows data to be stored and recovered asynchronously without the need for external buffering, complicated control programs, or the the appearance of large gap segments on the tape.

In the WRITE mode, the Model 173/133 accepts eight parallel binary bits, formats them, and writes them on tape.

In the READ mode it reads one bit at a time off the tape and presents it in parallel format at the output port.

Both the read and write operations are largely automatic and require minimal control signals. However, tape speed restrictions require certain timing considerations. Due to the duty cycle of the stepping motor, the START control pulses in the WRITE mode must be at least nine milleseconds apart. Otherwise, improper tape motion can occur and errors will result. In the READ mode the START pulses must be at least twelve milleseconds apart.

The input and output signals from the recorder are TTL compatible, with positive true logic. The input/output signals appear on the pins of a 25 pin male cannon connector. This connector also carries the control signals between the controlling computer and the Model 173/133. These signals and their respective pin numbers are specified in the Memodyne Recorder/Intellec 8 connection list in Appendix E.

A logical zero level on the LOAD FORWARD control line causes the tape to move off of the transparent leader and onto a portion of the tape where proper recording can proceed.

The EOT/BOT signal goes to a logical zero level when the tape is on the transparent leader in the cassette. A photo-diode can sense a small, internal lamp through the transparent leader and the sensor inhibits tape movement. This is to prevent moving off of the end of the tape and to prevent wearing of the pinch rollers when the tape is unable to move. The LOAD FORWARD signal overrides the inhibit condition.

The REWIND signal causes the tape to rewind to the leader. It is activated by a logical zero level on its line.

The START signal is a positive pulse which starts the stepping motor and enables the circuitry for READ/WRITE operations.

When the WRITE/READ line is brought to a logical zero level it causes the stepping motor to start on the arrival of a START pulse. The stepping motor will continue stepping until eight bits of data are written on the tape. A logical one level on this line will cause the recorder to read one data bit off of the tape when a START pulse arrives.

The TAPE SYNC line will go to a logical zero level when eight bits of data are ready at the output during a READ mode operation.

APPENDIX C

INTELLEC 8/MOD 8

The Intellec 8/MOD 8 is a stand alone computer manufactured by the INTEL Corporation. It is built around the INTEL 8008-1 microprocessor CPU. The Intellec 8 was designed primarily as a system development tool for INTEL 8008 microprocessor systems.

The Intellec 8 console contains power supplies, cabinet, and a display and control panel. The system has 8192 eight-bit bytes of random access memory, and 4096 bytes of programmable read only memory. The I/O module has four input ports, and four output ports. The console also contains a built-in PROM programmer, and a serial communications interface.

The software included in the Intellec 8 system consists of a resident System Monitor, Text Editor, and an Assembler.

The System Monitor program allows the user to load, manipulate, and dump the memory contents of the Intellec 8. It can also read and produce paper tapes, execute programs, and read or initialize PROM's.

Through the use of the input/output ports and the serial communications interface, the Intellec 8 can be connected with most standard peripherals to allow

input-output with teletype and paper punch consoles, video displays, paper tape readers, and magnetic tape recorders.

APPENDIX D

PL/M

PL/M is a high-level programming language that has been developed by the INTEL Corporation for system programming on the INTEL eight-bit microprocessor family of 8008 and 8080 CPU's.

PL/M is similar to PL/1 in that it is a highly structured language. This allows the user to develop programs as blocks, or procedures, which can then be combined into larger programs. Each procedure can be made independent of the rest of the program. In this way, the programmer can separately develop each part of the program and ensure that each portion is operating correctly.

PL/M greatly simplifies software development for system design on the 8008 and 8080. It provides automatic control of many of the processor's components and operations. Thus, the programmer does not have to keep track of and control all of the necessary register, memory, and stack operations required by the program. Yet, PL/M does allow the programmer specific control of logical byte manipulations, character string manipulations, and input/output operations.

PL/M is not as efficient in terms of register usage and memory requirements for program length as machine-language programming. But, PL/M is much easier to use for programming complex programs requiring mathematical

operations or large data array processing.

The PL/M Cross Compiler program is in the IBM -360 Program Library at the Naval Postgraduate School Computer Center and can be easily accessed through the CP/CMS operating system. The compiler provides symbol tables, source listings, and machine code listings. The machine code listing of the program can then be punched on paper tape, read into the Intellec 8, and run on the computer.

APPENDIX E

EQUIPMENT CONNECTIONS

A. INTELLEC 8

The intellec 8 has uncommitted input and output ports connected to two 37 pin connectors on the rear panel. These connectors are labeled as INPUT and OUTPUT. Matching 37 pin male cannon connectors can be used to interface external equipment to these ports. The connections used to interface the Intellec 8 with the data interface are as follows:

OUTPUT PORT (06)

Bit #	Pin #	Function
0	10	EXTERNAL ADDRESS ENABLE
1	11	No Connection
2	29	External Address Bit 9
3	30	Clock Flip-Flop Set Line
4	12	Clock Flip-Flop Reset Line
5	13	No Connection
6	31	No Connection
7	32	No Connection

OUTPUT PORT (07)

Bit #	Pin #	Function
0	14	External Address Bit 0
1	15	" " " 1
2	33	" " " 2
3	34	" " " 3
4	16	" " " 4
5	17	" " " 5
6	35	" " " 6
7	36	" " " 7

INPUT PORT (06)

Bit #	Pin #	Function
0	10	Data Bit 0
1	11	" " 1
2	29	" " 2
3	30	" " 3
4	12	" " 4
5	13	" " 5
6	31	" " 6
7	32	" " 7

INPUT PORT (07)

Bit #	Pin #	Function
0	14	Data Bit 8
1	15	" " 9
2	33	" " 10
3	34	" " 11
4	16	No Connection
5	17	No Connection
6	35	No Connection
7	36	No Connection

The Intellec 8 and the Memodyne recorder are connected through a twenty-five line conneting cable. The end of the cable connected to the Memodyne recorder uses a 25 pin male cannon connector. The connections to the Intellec are internal wiring connections. Input data from the Memodyne is connected to INPUT PORT (06) . This connection will be in parallel with the INPUT PORT (06) connections at the 37 pin cannon connector on the rear panel of the Intellec 8. Care must be taken so that both sets of connections are not being used at the same time. Connections from the Intellec 8 to the Memodyne Digital Cassette Recorder are:

INPUT PORT (06)

Bit #	Function
0	Recorder Input Bit # 0
1	" " " 1
2	" " " 2
3	" " " 3
4	" " " 4
5	" " " 5
6	" " " 6
7	" " " 7

OUTPUT PORT (05)

Bit #	Function
0	Cutput To Recorder Bit # 0
1	" " " " 1
2	" " " " 2
3	" " " " 3
4	" " " " 4

Bit #	Function
5	Output To Recorder Bit # 5
6	" " " " 6
7	" " " " 7

INPUT PORT (01)

Bit #	Function
1	TAPE SYNC Signal Input
2	EOT/BOT Signal Input

OUTPUT PORT (03)

Bit #	Function
0	LOAD FORWARD Signal
1	START Pulse
2	READ/WRITE Signal
3	FORWARD/REVERSE Signal

B. INTERFACE CIRCUIT BOARD

The interface circuit board uses a 22 pin card edge connector. Looking down on the component side of the board, with the pins toward the front, the pins are numbered from left to right. The connections are as follows:

Pin #	Function
1	Output Data Bit 11
2	" " " 10
3	" " " 9
4	" " " 8
5	" " " 7
6	" " " 6
7	" " " 5
8	" " " 4
9	" " " 3
10	" " " 2
11	" " " 1
12	" " " 0
13	No Connection
14	No Connection
15	No Connection
16	Vertical Data Input
17	Clock Flip-Flop Reset Input
18	Bit Clock input
19	Word Clock Input
20	Clock Flip-Flop Set Input
21	Ground
22	+5 Volts

C. SPECTRUM ANALYZER

The Analyzer has three rear panel multi pin connectors. The following signals present on the INPUT/OUTPUT connector J1 are connected to the Intellec 8 and the interface circuit board:

Pin #	Function
1	EXTERNAL ADDRESS ENABLE Input
4	EXTERNAL ADDRESS Input Bit 0
8	" " " 1
12	" " " 2
15	" " " 3
18	" " " 4
22	" " " 5
25	" " " 6
28	" " " 7
31	" " " 8
50	Vertical Data Output
53	Word Clock Output
57	Bit Clock Output

D. MEMODYNE CASSETTE RECORDER

The Memodyne recorder is connected to the Intellec 8 by a 25 pin male cannon connector. These connections are:

Pin #	Function
1	Ground
2	Input Bit 0
3	" " 1
4	" " 2
5	" " 3
6	" " 4
7	" " 5
8	" " 6
9	" " 7
10	LOAD FORWARD Input
11	No Connection
12	START Pulse Input
13	READ/WRITE Input
14	Output Bit 0
15	" " 1
16	" " 2
17	" " 3
18	" " 4
19	" " 5
20	" " 6
21	" " 7
22	EOT/BOT Output
23	TAPE SYNC Output
24	FORWARD/REVERSE Input
25	No Connection

```
 /*  PROCEDURE DELAY GIVES A PROGRAMMED DELAY IN
     INCREMENTS OF 25.0 MSEC.  THE TOTAL DELAY WILL
     BE AMOUNT X 25.0 MSEC.  IT IS INVOKED AS FOLLOWS:
     CALL DELAY(ARGUMENT)  */

DELAY: PROCEDURE(AMOUNT);
DECLARE (K,AMOUNT) ADDRESS;
DO K=1 TO AMOUNT;
   CALL TIME(250);
   END;
RETURN;
END DELAY;


 /*  PROCEDURE UN$HEXER TAKES A HEX NUMBER AS AN
     ARGUMENT, CONVERTS IT TO BINARY CODED DECIMAL
      (BCD), AND RETURNS THE BCD NUMBER TO THE
     CALLING PROGRAM.  IT IS INVOKED AS FOLLOWS:
     PLM$EXPRESSION = UN$HEXER(ARGUMENT).  THE
     ARGUMENT MUST BE LESS THAN 270F HEXADECIMAL  */

UN$HEXER:  PROCEDURE(HEX$NUMBER) ADDRESS;
DECLARE EGRESS LABEL;
DECLARE (HEX$NUMBER, BCD$NUMBER, TEMP) ADDRESS;
IF HEX$NUMBER < 10 THEN
   DO;
   BCD$NUMBER = HEX$NUMBER;
   GO TO EGRESS;
   END;
ELSE BCD$NUMBER = HEX$NUMBER MOD 10;
HEX$NUMBER = HEX$NUMBER/10;
IF HEX$NUMBER < 10 THEN
   DO;
   BCD$NUMBER = BCD$NUMBER OR SHL(HEX$NUMBER,4);
   GO TO EGRESS;
   END;
```

```
ELSE TEMP = HEX$NUMBER MOD 10;
BCD$NUMBER = BCD$NUMBER OR SHL(TEMP ,4);
HEX$NUMBER = HEX$NUMBER/10;
IF HEX$NUMBER < 10 THEN
   DO;
   BCD$NUMBER = BCD$NUMBER OR SHL(HEX$NUMBER,8);
   GO TO EGRESS;
   END;
ELSE TEMP = HEX$NUMBER MOD 10;
BCD$NUMBER = BCD$NUMBER OR SHL(TEMP,8);
HEX$NUMBER = HEX$NUMBER/10;
TEMP = SHL(HEX$NUMBER,8);
BCD$NUMBER = BCD$NUMBER OR SHL(TEMP,4);
EGRESS: RETURN BCD$NUMBER;
END UN$HEXER;


 /*  PROCEDURE MAG$TAPE$OUT TAKES AN ARRAY OF
    500 ADDRESS (TWO BYTES) VALUES, CONVERTS THEM
    TO BCD (USING PROCEDURE UN$HEXER), AND OUTPUTS
    THE ARRAY ON MAGNETIC TAPE.  IT IS INVOKED
    AS FOLLOWS:
    CALL MAG$TAPE$OUT (.DUMMY(1))
    .DUMMY(1) IS A POINTER TO THE FIRST ELEMENT
    OF THE ARRAY BEING PASSED.  */

MAG$TAPE$OUT:  PROCEDURE (VALUE$POINTER);
DECLARE OUT03 LITERALLY 'OUTPUT(03)';  /* (1) */
DECLARE OUT05 LITERALLY 'OUTPUT(05)';  /* (1) */
DECLARE VALUE$POINTER ADDRESS;
DECLARE VALUE BASED VALUE$POINTER ADDRESS;
DECLARE (I,BCD$VALUE) ADDRESS;
 /*  WRITE APPROXIMATELY TWELVE SECONDS OF
    BLANK TAPE.  THIS GAP IS TO ENSURE THAT
    THE TAPE IS OFF OF THE LEADER, AND SERVES
    TO SEPERATE MULTIPLE FILES ON THE TAPE.  */
```

```
OUT03 = 03H;  /* (2) */
CALL DELAY(500);
OUT03 = 02H;  /* (2) */
DO I=1 TO 500;
 /*  CONVERT THE VALUE TO BCD  */
   BCD$VALUE = UN$HEXER(VALUE);
 /*  OUTPUT HIGH ORDER BYTE  */
   OUT05 = HIGH(BCD$VALUE);  /* (2) */
 /*  START PULSE  */
   OUT03 = 00H;  /* (2) */
   OUT03 = 02H;  /* (2) */
   CALL DELAY(1);
 /*  OUTPUT LOW ORDER BYTE  */
   OUT05 = LOW(BCD$VALUE);  /* (2) */
 /*  START PULSE  */
   OUT03 = 00H;  /* (2) */
   OUT03 = 02H;  /* (2) */
   CALL DELAY(1);
 /*  INCREMENT POINTER TO ARRAY  */
   VALUE$POINTER = VALUE$POINTER + 2;
   END;
RETURN;
END MAG$TAPE$OUT;


 /*  PROCEDURE DATA$FETCH PROVIDES THE EXTERNAL
    CONTROL SIGNALS AND LOGIC THAT ARE NECESSARY
    TO INPUT THE SPECTRUM VALUES FROM THE ANALYZER
    AND PASSES THE ARRAY OF THESE VALUES TO PROCEDURE
    MAG$TAPE$OUT  */

DATA$FETCH:  PROCEDURE;
DECLARE OUT06 LITERALLY 'OUTPUT(06)';  /* (1) */
DECLARE OUT07 LITERALLY 'OUTPUT(07)';  /* (1) */
DECLARE IN06 LITERALLY 'NOT INPUT(06)';  /* (1),(3) */
DECLARE IN07 LITERALLY 'NOT INPUT(07)';  /* (1),(3) */
```

```
DECLARE EQ LITERALLY ' = NOT ';  /* DELETE NOT FOR (2) */
DECLARE BIN$VALUE(500) ADDRESS;
DECLARE (I,BIN$NUMBER) ADDRESS;
DECLARE TEMP BYTE;
 /*  INITIALIZE EXTERNAL ADDRESS INPUT TO 0000H  */
BIN$NUMBER = 0;
OUT06 EQ 02H;  /*  DISABLE EXT. ADV. PULSE
  ENABLE EXTERNAL ADDRESS  */
DO I=1 TO 500;
 /*  GET HIGH ORDER ADDRESS BYTE AND OUTPUT
    THE LOW ORDER BYTE  */
   TEMP = HIGH(BIN$NUMBER);
   TEMP = SHL(TEMP,2) AND 04H;
   OUT07 EQ LOW(BIN$NUMBER);
 /*  OUTPUT HIGH ORDER BYTE  */
   OUT06 EQ TEMP AND 06H;
   CALL TIME(5);
 /*  ENABLE THE DATA CLOCK */
   OUT06 EQ TEMP AND 0EH;
   CALL TIME(5);
 /*  DISABLE DATA CLOCK  */
   OUT06 EQ TEMP AND 16H;
   CALL TIME(5);
 /*  INPUT HIGH ORDER DATA BYTE  */
   BIN$VALUE(BIN$NUMBER) = IN07 AND 0FH;
   BIN$VALUE(BIN$NUMBER) = SHL(BIN$VALUE(BIN$NUMBER),8);
 /*  INPUT LOW ORDER DATA BYTE  */
   BIN$VALUE(BIN$NUMBER) = BIN$VALUE(BIN$NUMBER) OR IN06;
   BIN$NUMBER = BIN$NUMBER + 1;
   END;
CALL MAG$TAPE$OUT(.BIN$VALUE(0));
RETURN;
END DATA$FETCH;
CALL DATA$FETCH;  /*  DUMMY CALLING PROGRAM  */

EOF
```

/* PROGRAM TO PUNCH PAPER TAPE */

ADDRESS	CODE	LABEL	MNEMONIC	COMMENTS
2F00	2E	LINE	LHI	POINT MEMORY
01	2F			AT 2FF4
02	36		LLI	
03	F4			
04	3E		LMI	LOAD MEMORY WITH
05	0A			COUNTER VALUE '10'
06	0E		LBI	PUNCH A 'SPACE'
07	20			ASCII 'SPACE'
08	46		JSU	JUMP TO
09	B5			MONITOR PUNCH 'B'
0A	3E			ADDRESS
0B	0E		LBI	PUNCH ANOTHER
0C	20			'SPACE'
0D	46		JSU	
0E	B5			
0F	3E			
2F10	46	PAIR	JSU	FETCH HIGH BYTE
11	33			FETCH
12	2F			ADDRESS
13	46		JSU	PUNCH HIGH BYTE
14	7B			MONITOR PUNCH
15	3E			ADDRESS
16	46		JSU	FETCH LOW BYTE
17	33			FETCH
18	2F			ADDRESS
19	46		JSU	PUNCH LOW BYTE
1A	7B			MONITOR PUNCH
1B	3E			ADDRESS
1C	2E		LHI	TEN NUMBERS
1D	2F			PUNCHED YET?
1E	36		LLI	POINT MEMORY
2F1F	F4			AT COUNTER

ADDRESS	CODE	LABEL	MNEMONIC	COMMENTS
2F20	CF		LBM	LOAD COUNTER IN B
21	09		DCB	DECREMENT COUNTER
22	48		JFZ	NO, SAVE COUNTER
23	2B			RESTORE
24	2F			ADDRESS
25	46		JSU	YES, GO X OFF
26	5C			X OFF
27	2F			ADDRESS
28	44		JMP	GO TO LINE AGAIN
29	00			LINE
2A	2F			ADDRESS
2B	2E	RESTORE	LHI	POINT MEMORY
2C	2F			AT 2FF4
2D	36		LLI	
2E	F4			
2F	F9		LMB	STORE COUNTER
2F30	44		JMP	GET NEXT NUMBER
31	06			PAIR
32	2F			ADDRESS
33	2E	FETCH	LHI	POINT MEMORY
34	2F			AT 2FF0
35	36		LLI	
36	F0			
37	CF		LBM	LOAD B AND C
38	30		INL	WITH DATA ADDRESS
39	D7		LCM	
3A	E9		LHB	POINT MEMORY
3B	F2		LLC	AT DATA
3C	DF		LDM	DATA IN D
3D	30		INL	INCREMENT ADDRESS
3E	48		JFZ	END OF PAGE?
2F3F	42			NO, GO TO STORE

ADDRESS	CODE	LABEL	MNEMONIC	COMMENTS
2F40	2F			
41	28		INH	YES, INCREMENT PAGE
42	CD	STORE	LBH	LOAD B AND C WITH
43	D6		LCL	NEXT ADDRESS
44	2E		LHI	POINT MEMORY
45	2F			AT 2FF0
46	36		LLI	
47	F0			
48	F9		LMB	STORE NEXT ADDRESS
49	30		INL	
4A	FA		LMC	
4B	30		INL	LOAD A WITH THE HI
4C	C7		LAM	BYTE OF TOP ADDRESS
4D	B9		CPB	COMPARE WITH HI BYTE
4E	48		JFZ	NOT DONE, GO TO DATA
4F	5A			DATA
2F50	2F			ADDRESS
51	30		INL	LOAD A WITH THE LOW
52	C7		LAM	BYTE OF TOP ADDRESS
53	BA		CPC	COMPARE WITH LOW
54	48		JFZ	BYTE AND GO TO DATA
55	5A			IF NOT DONE
56	2F			
57	44		JMP	IF DONE, GO STOP
58	6A			STOP
59	2F			ADDRESS
5A	C3	DATA	LAD	DATA IN A
5B	07		RET	RETURN
5C	0E	X OFF	LBI	PUNCH X OFF
5D	13			ASCII 'X OFF'
5E	46		JSU	
2F5F	B5			MONITOR PUNCH 'R'

ADDRESS	CODE	LABEL	MNEMONIC	COMMENTS
2F60	3E			ADDRESS
61	03		LBI	PUNCH A 'SPACE'
62	20			ASCII 'SPACE'
63	46		JSU	
64	B5			MONITOR PUNCH 'B'
65	3E			ADDRESS
66	C0		NOP	A CRLF CAN BE
67	C0		NOP	INSERTED HERE AS A
68	C0		NOP	46 95 3E IF DESIRED
69	07		RET	RETURN
6A	0E	STOP	LBI	PUNCH THREE SETS OF
6B	13			X OFF TO MARK EOF
6C	46		JSU	
6D	B5			
6E	3E			
6F	0E		LBI	
2F70	13			ASCII 'X OFF'
71	46		JSU	
72	B5			
73	3E			
74	0E		LBI	
75	13			ASCII 'X OFF'
76	46		JSU	
77	B5			
78	3E			
79	44		JMP	GO TO MONITOR
7A	00			MONITOR
2F7B	38			ADDRESS

LIST OF REFERENCES

1. Federal Scientific Corporation, Instruction Manual for MODEL UA-500-1 Analyzer Averager, 1973.

2. Teletype User's Guide for CP/CMS and Processing Paper Tape Data Under CP/CMS, Naval Postgraduate School Technical Note No. 0211-18, April 1974.

3. Intel Corporation, Intellec 8 Reference Manual, Rev. 1, 1974.

4. Intel Corporation, 8008 and 8080 PL/M Programming Manual, Rev. A, 1975.

5. Memodyne Corporation, Technical Manual 100 Series Tape Transports and Recording Systems, 1972.

6. Memodyne Corporation, Instruction Manual, Model 173, 1972.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Documentation Center Cameron Station Alexandria, Virginia 22314	2
2.	Library, Code 0212 Naval Postgraduate School Monterey, California 93940	2
3.	Department Chairman, Code 67 Department of Aeronautics Naval Postgraduate School Monterey, California 93940	1
4.	Dr. Thomas M. Houlihan, Code 69Hm Associate Professor, Department of Mechanical Engineering Naval Postgraduate School Monterey, California 93940	3
5.	LT Thomas F. Sauntry, USN 4822 East Paiute St. Phoenix, Arizona 85044	1